

notification to the software application **106b** on node **104b**. The software application **106b** can access the lookup table **112b** to determine the mitigation operation for the event notification. Similar steps can be performed with respect to the software application **106c**, monitoring agent **108c**, the notification controller **110**, and the lookup table **112c**.

[0021] FIG. 2 is a block diagram of an example of a system **200** for implementing a resource-usage notification framework in a distributed computing environment **102** according to some aspects. The system **200** includes a processor **202** communicatively coupled with a memory **204**. In some examples, the processor **202** and the memory **204** can be parts of one or more nodes **104a-c** of the distributed computing environment **102** of FIG. 1.

[0022] The processor **202** can include one processor or multiple processors. Non-limiting examples of the processor **202** include a Field-Programmable Gate Array (FPGA), an application-specific integrated circuit (ASIC), a microprocessor, etc. The processor **202** can execute instructions **206** stored in the memory **204** to perform operations. In some examples, the instructions **206** can include processor-specific instructions generated by a compiler or an interpreter from code written in any suitable computer-programming language, such as C, C++, C#, etc.

[0023] The memory **204** can include one memory or multiple memories. The memory **204** can be non-volatile and may include any type of memory that retains stored information when powered off. Non-limiting examples of the memory **204** include electrically erasable and programmable read-only memory (EEPROM), flash memory, or any other type of non-volatile memory. In some examples, at least some of the memory can include a medium from which the processor **202** can read instructions **206**. A computer-readable medium can include electronic, optical, magnetic, or other storage devices capable of providing the processor **202** with computer-readable instructions or other program code. Non-limiting examples of a computer-readable medium include magnetic disk(s), memory chip(s), ROM, random-access memory (RAM), an ASIC, a configured processor, optical storage, or any other medium from which a computer processor can read the instructions **206**.

[0024] In some examples, the processor **202** determines resource usage **210** by a software application **106** in a distributed computing environment **102**. The processor **202** can determine that the resource usage **210** is within a predefined range of a predefined resource-consumption limit **212**. Based on determining that the resource usage **210** is within the predefined range of the predefined resource-consumption limit **212**, the processor **202** generates an event notification **214**. The processor **202** can transmit (e.g., directly or indirectly) the event notification **214** to the software application **106** for causing the software application **106** to receive the event notification **214** and responsively perform a mitigation operation **216**. The mitigation operation **216** can be configured to prevent the resource usage **210** from exceeding the predefined resource-consumption limit **212** or mitigate an impact of the resource usage **210** exceeding the predefined resource-consumption limit **212**. For example, the mitigation operation **216** can involve storing intermediate results of a data-processing operation implemented by the software application **106**, so as to mitigate an impact of the software application **106**

being killed as a result of the software application's resource usage approaching or exceeding the predefined resource-consumption limit **212**.

[0025] In some examples, the processor **202** can implement some or all of the steps shown in FIG. 3. Other examples can include more steps, fewer steps, different steps, or a different order of the steps than is shown in FIG. 3. The steps of FIG. 3 are discussed below with reference to the components discussed above in relation to FIG. 2.

[0026] In block **302**, the processor **202** determines resource usage **210** by a software application **106** in a distributed computing environment **102**. The resource usage **210** can include memory usage, disk usage, network usage, or processing-unit usage. The processor **202** can communicate with a monitoring agent to determine the resource usage **210**.

[0027] In block **304**, the processor **202** determines that the resource usage **210** is within a predefined range of a predefined resource-consumption limit **212**. An administrator or operator of the distributed computing environment **102** may set the predefined resource-consumption limit, in some examples. If resource usage **210** is within the predefined range of the predefined resource-consumption limit **212**, it can indicate a high resource-consumption event.

[0028] In block **306**, the processor generates an event notification **214** based on determining that the resource usage **210** is within the predefined range of the predefined resource-consumption limit **212**. The event notification **214** can indicate a type of the high resource-consumption event, such as a high memory-consumption event, a high disk-usage event, a high processing-usage event, a high network-usage event, or any combination of these. This may allow the software application **106** to perform one or more different types of mitigation operations depending on the one or more types of high resource-consumption events.

[0029] In block **308**, the processor **202** transmits the event notification **214** to the software application **106** for causing the software application **106** to receive the event notification **214** and responsively perform a mitigation operation **216**. In some examples, the mitigation operation **216** can be configured to prevent the resource usage **210** from exceeding the predefined resource-consumption limit **212**. In other examples, the mitigation operation **216** can be configured to mitigate an impact of the resource usage **210** exceeding the predefined resource-consumption limit **212**. The software application **106** can be configured to determine the mitigation operation **216** in response to the event notification **214** by accessing a lookup table that specifies correlations between event notifications and mitigation operations.

[0030] The foregoing description of certain examples, including illustrated examples, has been presented only for the purpose of illustration and description and is not intended to be exhaustive or to limit the disclosure to the precise forms disclosed. Numerous modifications, adaptations, and uses thereof will be apparent to those skilled in the art without departing from the scope of the disclosure. For instance, any example(s) described herein can be combined with any other example(s) to yield further examples.

1. A system comprising:
  - a processor; and
  - a memory including instructions that are executable by the processor for causing the processor to:
    - determine resource usage by a software application in a distributed computing environment;